REMARKS

Claim 34 has been amended to more clearly capture the claimed invention. Claims 1-35 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Priority:

Applicant notes that an erroneous priority claim to a provisional application was inadvertently included in the declaration for the instant application. No priority claim to any provisional or non-provisional application was intended. Please ignore the incorrect priority claim to provisional application 60/286,407 in the declaration.

Drawing Objections:

Figures 6B and 8A have been corrected and Applicant submits that the Examiner's objections to the drawings have now been overcome. Two replacement sheets are included herewith.

Specification Objections:

The Specification has been corrected to overcome the Examiner's objections.

Section 102(b) Rejection:

The Office Action rejected claims 1, 3-5, 24, 25, 27-29 and 34 under 35 U.S.C. § 102(b) as being anticipated by Kurihara (U.S. Patent 4,107,649). Applicant respectfully traverses this rejection for at least the reasons presented below and submits that Kurihara does not anticipate claims 1, 3-5, 24, 25, 27-29 and 34.

Regarding claim 1, contrary to the Examiner's assertion, Kurihara does not disclose shifting a first bit having a different logical value than the same logical value across the initial data bit combination, wherein each time the first bit is shifted, one of n data bit combinations is generated. In contrast, Kurihara teaches a system in which a shift register sequentially shifts input data and feeds back output data and in which a parity signal is generated and compared to predicted, or counted, parity signal to determine whether the shift register is working correctly. Specifically, Kurihara teaches that input data is shifted through a shift register while the number of logical values of '1' in the input data are counted and the number of logical values of '1' outputted by the shift registers is also counted (Kurihara, column 3, lines 13-24). Additionally, Kurihara's shift register performs a parity calculation on the data in the shift register. Kurihara then uses the number of 1s counted in the input data and/or the number of 1s counted in the output data to predict the parity value for the data in the shift register (Kurihara, column 3, lines 25-38). Kurihara's system includes a check circuit to compare the predicted or counted parity value with the parity value generated by the shift register and generates an error signal if they do not match (Kurihara, column 4, lines 13-19). However, Kurihara does not teach shifting a first bit having a different logical value across the initial data bit combination, wherein each time the bit is shifted, one of n data bit combinations is generated. While Kurihara does use a shift register and does shift input data through the shift register, Kurihara does not teach generating one of n data bit combinations each time the first bit is shifted. Kurihara only teaches that input data is shifted in to the shift register.

In further regard to claim 1, Kurihara fails to teach providing each of the n data bit combinations to the error detection/correction logic; in response to said providing, the error detection/correction logic generating a set of check bits for each of the n data bit combinations; comparing the set of check bits generated by the error correction/detection logic with a known correct set of check bits for each of the n data bit combinations. Instead, Kurihara teaches only that input data enters the input end of the shift register (Kurihara, column 3, lines 13-15). Kurihara mentions nothing about providing each of n data bit combinations to the error detection/correction logic. Kurihara does not mention anything about the size of input data nor about the input data including n data bit combinations. Kurihara is equally silent regarding the error detection/correction logic generating a set of check bits. Instead, Kurihara's shift register calculates the single parity bit from the input data in the shift register. Calculating a single parity bit is very different from generate a *set of check bits*. Kurihara also does not disclose comparing the set of check bits generated by the error correction/detection logic with a known correct set of check bits for each for the n data bit combinations. In contrast, Kurihara compares a generated parity bit with a predicted, or counted, parity bit from the input data (and/or output data). The counted parity value of Kurihara is clearly not a *known correct set of check bits*. Furthermore, Kurihara does not teach comparing the set of check bits with a known correct set of check bits *for each of the n data bit combinations*. Kurihara only teaches that input data is shifted into the shift register and that the number of logical values of 1 in the input data are counted. Kurihara does not mention n data bit combinations nor does he disclose comparing check bits with known correct check bits *for each of the n data bit combinations*.

Kurihara fails to anticipate Applicants claim 1 and thus the rejection of claim 1 is not supported by the prior art and removal thereof is respectfully requested.

In regards to claim 24, Kurihara fails to teach test check bit generating means for creating a set of test data bit combinations and providing the set of test data bit combinations to a check bit generator comprised in the error correction/detection logic. In contrast, as discussed above, Kurihara teaches inputting data to a shift register. Kurihara does not disclose creating a set of test data bit combination and providing the set of test data bit combinations to a check bit generator. The Examiner cites FIG 2 and column 5, lines 7-34 of Kurihara. The cited portions of Kurihara only describe shifting input data into a shift register, generating a parity value for the input data in the shift register and counting the values of 1 in the input data for comparison. The Examiner appears to be implying that Kurihara's input data is a set of test data bit combinations and that Kurihara's shift register is a check bit generator. However, Kurihara's does not teach that the input data to the shift register is a set of test data bit combinations created by test check bit generating means.

Further regarding claim 24, Kurihara also fails to disclose wherein the set of test data bit combinations comprises n n-bit data bit combinations, wherein each possible value of each data bit is present in at least one of the n n-bit data bit combinations in the set of test data bit combinations. Nowhere does Kurihara mention anything about a set of test data bit combinations comprising n n-bit data bit combinations and the Examiner's cited passages (FIG 2 and column 5, lines 7-34) only describe shifting input data into a shift register, generating a parity value for the data in the shift register and counting the values of 1 in the input data for comparison. Neither the cited portions, nor the remainder of Kurihara, mention anything regarding n n-bit data combinations. Furthermore, nowhere does Kurihara teach that each possible value of each data bit is present in at least one of the n n-bit data bit combinations of test data bit combinations. Kurihara only refers generically to "input data" without describing anything about any combinations of bits in the makeup of the input data (see, Kurihara, column 1, lines 44-48, and column 3, lines 13-17).

Additionally, Kurihara fails to teach comparison means for comparing check bits output by the error/detection logic for each of the n n-bit data bit combinations in the set of test data bit combinations to known correct check bits for each of the n n-bit data bit combinations. Kurihara only teaches comparing a one-bit predicted parity value, generated by counting the "1"s in input data, with a one-bit parity value generated on the data in the shift register. Nowhere does Kurihara disclose anything regarding comparing check bits for each of n n-bit data bit combinations. As noted above, Kurihara fails to mention n n-bit data bit combinations at all.

Applicants respectfully remind the Examiner that Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim. *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 221 USPQ 481, 485 (Fed. Cir. 1984). The identical invention must be shown in as complete detail as is contained in the claims. *Richardson v. Suzuki Motor Co.*, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). Kurihara clearly does not anticipate Applicants claim 24. Thus, the rejection of claim 24 is not supported by the prior art and removal thereof is respectfully requested.

Regarding claim 27, in contrast to the Examiner's contention, Kurihara does not teach a method including providing a set of m+1 test code words to the error correction/detection logic, wherein each code word has m bits, wherein a first test code word in the set of m+1 test code words is a correct code word, wherein each test code word other than the first test code word comprises a single-bit error at a different bit position within the code word than each other test code word. The Examiner's cited portions (FIG 2, column 2, lines 61-68 and column 3, lines 13-34) only describe how Kurihara calculates a predicted parity value and compares it to a generated parity value to detect malfunction of the error detection circuit. However, nowhere in the cited passages does Kurihara mention providing a set of m+1 test code words to the error correction/detection logic. The Examiner is apparently equates Kurihara's references to "input data" shifted into a shift register as teaching providing a set of m+1 test code words to the error correction/detection logic. However, Kurihara never mentions a set of m+1 test code words, nor does he describe his "input data" as test code words. Further, Kurihara does not describe his input data as having m bits. In fact, Kurihara never mentions anything about the size of input data. Thus, Kurihara's input data cannot be equated to a set of m+1 test code words, wherein each code word has m bits, as the Examiner contends.

Additionally, Kurihara fails to disclose wherein a first test code word in the set of m+1 test code words is a correct code word, wherein each test code word other than the first test code word comprises a single-bit error at a different bit position within the code word than each other test code word. Kurihara's method simply does not include a set of m+1 test code words that includes a first test code word that is a correct code words and wherein the other test code words each include a single-bit error at different bit positions. Kurihara teaches completely different type of method incompatible with a set of test code words. Kurihara's entire method is only concerned with comparing predicted and generated parity values. It would not even make since to use a set of m+1 test code words where every code word other than a first correct code word includes single bit errors in Kurihara's method because his method would merely compare the predicted parity value (total counted "1"s) with a generated parity value (by a parity generator). Such a method has no way to detect or make use of single-bit errors in test code words.

Furthermore, Kurihara's method clearly does not include the error correction/detection logic decoding the set of m+1 test code words. As noted above, Kurihara does not teach providing a set of m+1 test code words to the error correction/detection logic. Kurihara also fails to teach error correction/detection logic that decodes such a set of m+1 test code words. Kurihara's entire system includes only a shift register, 2 one-bit counters, a parity generator, and a check circuit (FIGs 1, 2, and column 3, lines 39-65). There is nothing in Kurihara's system that is described as capable of decoding a set of m+1 test code words.

Kurihara clearly fails to anticipate claim 27 and Applicants respectfully request removal of the rejection of claim 27 for at least the reasons give above.

Regarding amended claim 34, Kurihara does not teach a system that includes a storage array comprising at least one mass storage device. The Examiner cites column 3, lines 13-24 of Kurihara that describes how Kurihara calculates a predicted parity value and compares it to a generated parity value to detect malfunction of the error detection circuit. The Examiner's other cited passage (column 5, line 52-column 6, line 14) refers to data processing circuit including a shift register and also having feedback means, parity generator means, predicting means and comparing means. Nothing in either of the cited passages can be considered a storage array comprising at least one mass storage device. A shift register is clearly not a mass storage device or a storage array comprising a mass storage device.

Further regarding claim 34, Kurihara does not teach a system including error correction/detection logic configured to generate check bits for data being provided to a storage array comprising at least one mass storage device. Kurihara teaches a shift register and parity generator neither of which generates check bits for data being provided to a storage array comprising a mass storage device. Kurihara does not even mention a storage array and also fails to teach generating check bits for data being provided to a storage array. Kurihara's system is not concerned with generated check bits for data being provided to a storage array. Kurihara's system is concerned with a "circuit for checking an error detection circuit for use in a data processing circuit" (Kurihara, column 1, lines 39-43).

Also, Kurihara fails to teach providing each of a set of n data bit combinations to the error detection/correction logic. Kurihara mentions nothing about providing each of n data bit combinations to the error detection/correction logic. Kurihara does not mention anything about the size of input data nor does he mention input data including a set of n data bit combinations. The Examine has not cited any passage of Kurihara that discusses a set of n data bit combinations. Additionally, remarks presented above regarding claim 1 apply to claim 34 as well.

Additionally, Kurihara does not teach wherein the host computer system is configured to test the error correction/detection logic by providing each of a set of n data bit combinations to the error detection/correction logic and wherein the host computer system is configured to compare the set of check bits generated by the error correction/detection logic with a known correct set of check bits for each of the n data bit combinations. In contrast, Kurihara teaches that check circuit 5 compares the parity value generated by parity generator 4 with the predicted parity value generated by counters 3 and/or 4 (Kurihara, FIG 1, and column 3, lines 25-34). However, check circuit 5 does not provide any data to Kurihara's shift register and certainly does not provide a set of n data bit combinations to error detection/correction logic. Furthermore, Kurihara's check circuit 5 cannot be considered a host computer system.

Kurihara clearly does not anticipate Applicants' claim 34. Thus, the rejection of claim 34 is not supported by the prior art and removal thereof is respectfully requested.

The Office Action rejected claim 30 under 35 U.S.C. § 102(b) as being anticipated by Arroyo, et al. (U.S. Patent 5,502,732) (hereinafter "Arroyo"). Applicants respectfully traverse this rejection for at least the reasons given below.

Regarding claim 30, contrary to the Examiner's assertion, Arroyo does not teach a method including providing a set of test code words to the error correction/detection logic, wherein said providing comprises introducing an error into each of the test code words in the set by substituting check bits corresponding to an unused syndrome for a correct set of check bits within each test code word, wherein each test code word comprises substituted check bits corresponding to a different unused syndrome than each other test code word in the set of test code words. Arroyo teaches a system for checking the test logic contained in a computer memory system during POST (Abstract). The Examiner's cited passage (Arroyo, column 5, line 55-column 6, line 4) describes how a CPU writes data including two bits set to "1" to memory and how diagnostic bit controlling multiplexer 33 replaces the two bits set to "1" with zeros. In Arroyo's system ECC generator 31 generates check bits for the unmodified data and stores those check bits in memory. When the data is read from memory (including the zeros substituted by multiplexer 33), ECC generator 41 generates check bits that does not match the check bits originally generated by ECC generator 31 when the data was stored. Hence, Arroyo does not substitute check bits corresponding to an unused syndrome, but rather teaches replacing two bits set to "1" with bits set to "0". These are two very different techniques.

Further, Arroyo's method does not include providing a set of test code words, wherein each test code word comprises substituted check bits corresponding to a different unused syndrome. In contrast, Arroyo teaches that "a multiplexer is provided in the data write path which substitutes *a constant set of identical bits* for the actual data generated by the CPU" (emphasis added, Arroyo, column 2, lines 22-25). Arroyo clearly teaches substituting a constant set of identical bits and thus clearly teaches away from providing a set test code words wherein each test code word comprises substituted check bits corresponding to a different unused syndrome.

For at least the reasons given above, the rejection of claim 30 is not supported by the prior art and removal thereof is respectfully requested.

Section 103(a) Rejection:

The Office Action rejected claim 2 under 35 U.S.C. § 103(a) as being unpatentable over Kurihara in view of Nielson, et al. (U.S. Patent 5,619,642) (hereinafter "Nielson"), claims 6-12, 26, 31-33 and 35 as being unpatentable over Kurihara as applied to claim 3 above, and further in view of Arroyo, claims 13-16 as being unpatentable over Kurihara in view of Fielder, et al. (U.S. Patent 6,446,037) (hereinafter "Fielder"), claims 17-23 as being unpatentable over Kurihara and Fielder and further in view of Arroyo. Dependent claims 2, 6-12, 26, 31-33, and 14-23, as being dependent upon claims that have been shown above to patentably distinguish of the prior art. Thus, Applicants respectfully traverse the rejections of claims 2, 6-12, 26, 31-33, and 14-23, in light of their respective independent claims.

Regarding claim 35, Kurihara in view of Arroyo fails to teach providing a subset of possible data bit combinations of n data bits to the error detection/correction logic, wherein the subset comprises n data bit combinations, wherein each possible value of each data bit is present in at least one of the n data bit combinations in the subset. As noted above regarding claim 1, Kurihara teaches comparing a one bit predicted parity value with a one bit generated parity value. Arroyo teaches substitutes "a constant set of identical bits [zeros]" (Arroyo, column 2, lines 22-25). Neither Kurihara nor Arroyo teach providing a subset of data bit combinations of n data bits, wherein each possible value of each data bit is present in at least one of the n data bit combinations. Kurihara only teaches input data without mentioning anything regarding possible data bit combinations of n data bits and Arroyo teaches the use of a constant set of identical bits.

Additionally, Kurihara in view of Arroyo also fails to teach verifying the error detection/correction logic by comparing a set of check bits generated by the error detection/correction logic for each of the n data bit combinations in the subset with a set of known correct check bits. The Examiner contends that Kurihara teaches this. However, the Examiner's cited portions of Kurihara (FIG 2, column 1, lines 44-60 and column 2, lines 12-19) only refer to comparing a predicted parity value with a generated parity value for input data in a shift register. Nowhere does Kurihara mention anything comparing a set of check bits generated by the error detection/correction logic for each of n data bit combinations. In fact, as noted above, Kurihara fails to mention n data bit combinations as all and certainly does not disclose comparing a set of check bits generated for each of n data bit combinations with a set of known correct check bits. Arroyo also fails to teach such functionality. In contrast, as noted above, Arroyo teaches substituting a constant set of identical bits (zeros) when writing data to memory and comparing check bits generated on both the original data and the data with zeros substituted. Since neither Kurihara, nor Arroyo teaches comparing check bits generated for each of n data bit combinations with a set of known correct check bits, no combination of Kurihara and Arroyo would include such a feature.

Kurihara in view of Arroyo also fails to teach providing a first set of m+1 test code words to the error detection/correction logic, wherein a first test code word is a correct test code word and where each other test code word in the set of m+1 test code words comprises a single-bit error, wherein each test code word having a single-bit error has the single-bit error at a different bit position than each other test code word that has a single-bit error. The Examiner asserts that Kurihara teaches such functionality in his method and cites Fig 2, column 2, lines 61-68 and column 3 lines 13-34. However, as noted above regarding the rejection of claim 27, these cites portions fail to mention anything about a set of m+1 test code words and the remarks and arguments presented above regarding claim 27 also apply here. Arroyo also fails to teach providing a set of m+1 test code words wherein a first test code word is a correct test code word and where each other test code word comprises a single bit error at a different bit offset than each other test code word. As noted above, Arroyo teaches that "a multiplexer is provided in the data write path which substitutes *a constant set of identical bits* for the actual data generated by the CPU" (emphasis added, Arroyo, column 2, lines 22-25). Arroyo further teaches an ECC generator 31 that generates check bits for the unmodified data and stores those check bits in memory. When the data is read from memory (including the zeros substituted by multiplexer 33), ECC generator 41 generates check bits that does not match the check bits originally generated by ECC generator 31 when the data was stored. Hence, Arroyo does not provide a set of m+1 test code words, but rather teaches comparing check bits generated for a set of data with check bits generated on the same data after having zero's inserted for certain bits. Thus, Arroyo clearly fails to teach providing a set of m+1 test code words wherein a first test code word is a correct test code word and where each other test code word comprises a single bit error at a different bit offset than each other test code word. Additionally, no combination of Kurihara and Arroyo can include such a feature.

Kurihara in view of Arroyo also fails to teach providing a second set of test code words to the error detection/correction logic, wherein each test code word in the second set comprises an error introduced by substituting check bits corresponding to an unused syndrome for a correct set of check bits within a correct code word, wherein each test code word in the second set comprises substituted check bits corresponding to a different unused syndrome than each other test code word in the second set of test code words. Kurihara clearly fails to teach substituting check bits corresponding to an unused syndrome and the Examiner relies upon Arroyo for this feature. However, Arroyo also fails to teach substituting check bits corresponding to an unused syndrome. Please see above regarding claim 30 for remarks regarding how Arroyo fails to teach substituting check bits corresponding to an unused syndrome.

Therefore the rejection of claim 35 is clearly not supported by the prior art and removal thereof is respectfully requested.

Regarding claim 13, Kurihara in view of Fielder fails to teach a computer readable medium comprising program instructions computer executable to: create an initial data bit combination having n bits, wherein each data bit in the initial data bit combination has a same logical value as each other data bit in the initial data bit combination; shift a first bit having an different logical value than the same logical value across the initial data bit combination, wherein each time the first bit is shifted, one of n data bit combinations is generated; provide each of the n data bit combinations to error detection/correction logic; compare a set of check bits generated by the error correction/detection logic with a known correct set of check bits for each of the n data bit combinations; and dependent on an outcome of said comparing, generate an indication of whether the error detection/correction logic correctly generated the set of check bits. Please note that the remarks above regarding claim 1 in view of Kurihara also apply here to claim 13.

Additionally, Fielder teaches a method for scalable coding of audio data into a core layer in response to a desired noise spectrum established according to psychoacoustic principles that supports coding augmentation data into augmentation layers in response to various criteria (Fielder, Abstract). Fielder's method has nothing to do with the error detection/correction circuit of Kurihara. The Examiner is relying upon Fielder to teach a computer readable medium. The respective inventions of Kurihara and Fielder are directed to completely different fields of endeavor. Applicant can imagine no conceivable way to combine Kurihara's error detection circuit checking circuit with Fielder's scalable audio encoding process. Kurihara teaches a hardware circuit for detecting malfunction of an error detection circuit and does not teach anything related to Fielder's scalable audio encoding software. The Examiner's cited passage from Fielder (column 4, lines 60-64) describes how Fielder's scalable audio encoding and decoding processes may be conveyed by a machine readable medium. However, Kurihara teachings are specific to a hardware circuit implementation. There is no suggestion to in either Fielder or Kurihara to create a program instruction computer readable medium implementation of Kurihara's circuit. In fact, since Kurihara teachings are specific to a hardware circuit, it is unclear how Kurihara's teachings could even be applied to a program instruction computer readable medium implementation.

Moreover, the Fielder reference is not analogous art. "In order to rely on a reference as a basis for rejection of an applicant's invention, the reference must either be in the field of applicant's endeavor or, if not, then be reasonably pertinent to the particular problem with which the inventor was concerned." *In re Oeticker*, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1445 (Fed. Cir. 1992). "A reference is reasonably pertinent if, even though it may be in a different field from that of the inventor's endeavor, it is one which, because of the matter with which it deals, logically would have commended itself to an inventor's attention in considering his problem." *In re Clay*, 966 F.2d 656, 659, 23 USPQ2d 1058, 1060-61 (Fed. Cir. 1992). Here, Fielder is clearly not in the field of Applicant's endeavor of testing error correction/detection logic. In contrast, Fielder deals with scalable encoding and decoding of audio data (Fielder, col. 2, lines 35-49). Furthermore, the subject of Fielder would not logically have commended itself to an inventor's attention when considering the problem addressed by Applicant. One of skill in the art seeking to address the problem of testing error correction/detection logic would not have any logical reasons for considering a technique used to encode and decode audio data. Thus, Fielder is not within Applicant's field of endeavor and is not pertinent to the problem addressed by Applicant's invention. Not is Fielder within the field of endeavor or pertinent to the problem addressed by Kurihara. Accordingly, Fielder is non-analogous art and cannot properly be combined with Kurihara.

In the Office Action, the Examiner states that Fielder is in an analogous art. The Examiner has clearly over-generalized the meaning of "analogous art." *In re Oeticker* refers to Applicant's field of endeavor and particular problem. The analogous art requirement can always be made meaningless by over-generalizing the field of endeavor or problem. Almost any art may be considered pertinent if the problem is stated in general enough terms. That is why the courts have insisted that art used in § 103 rejections be in the same field of endeavor or pertinent to the particular problem. Fielder pertains to scalable encoding and decoding of audio data which has nothing to do with Applicant's field of endeavor or particular problem.

Therefore, for at least the reasons given above, the rejection of claim 13 is not supported by the prior art and removal thereof is respectfully requested.

Applicant also asserts that numerous ones of the dependent claims recited further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

CONCLUSION

Applicants submit the application is in condition for allowance, and notice to that effect is respectfully requested.

If any extension of time (under 37 C.F.R. § 1.136) is necessary to prevent the above referenced application from becoming abandoned, Applicant hereby petitions for such extension. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-94400/RCK.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard

□ Petition for Extension of Time

□ Notice of Change of Address

□ Fee Authorization Form authorizing a deposit account debit in the amount of $ for fees ( ).

☒ Two Replacement Drawing Sheets

Respectfully submitted,

Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: October 27, 2004

In the Drawings:

Please amend FIG. 6B to change "syndrome 18" to "syndrome 16" and amend FIG. 8A to change "Error Detection/Correction Unit 306" to "Error Detection/Correction Unit 306A." Replacement sheets are attached hereto for drawings sheets including these FIG. 6B and FIG. 8A respectively.